Quando um bloco começa a ser executado, as variáveis declaradas dentro dele são ativadas (passam efetivamente a existir) e permanecem ativas até o término de execução do bloco. Em outras palavras uma variável declarada no início de um bloco só existe dentro dele. Diz-se então que as variáveis são *locais* ao bloco, o que permite que elas só ocupem memória quando necessário.

Com isto, o programador pode gerenciar a utilização da memória. Por exemplo, um vetor ou matriz cuja dimensão não é conhecida *a priori* (p. ex., vai ser lida ou depende de algum processamento), pode ser declarado dentro de um bloco e, portanto, alocado dinamicamente na memória.

Exemplo

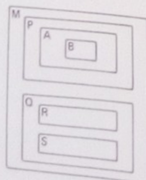
```
início
   inteiro N, I;
   leia (N);
   início
      tipo v = vetor [1:N] real;
      v: VET;
      para I de 1 até N faça
         VET [I] ← I;
      fim para;
      imprima (VET);
   fim;
fim.
```

7.2 ESCOPO DE VARIÁVEIS

Diz-se que um bloco é externo a outro, quando o segundo faz parte do primeiro.

Neste sentido, uma variável declarada em um bloco é *global* para todos os blocos internos e *local* para o próprio bloco.

Podemos ter diversos blocos aninhados, conforme mostra a figura a seguir.



nível	*bloco*
0	*M*
1	*P, Q*
2	*A, R, S*
3	*B*

P e *Q* são internos a *M* — *M* é externo a *P* e *Q*
A é interno a *P* — *P* é externo a *A*
B é interno a *A* — *A* é externo a *B*
R e *S* são internos a *Q* — *Q* é externo a *R* e *S*

Conforme foi mostrado, uma variável declarada dentro de um bloco só é conhecida dentro deste bloco.

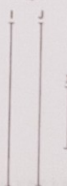
Se uma variável *A* declarada em um bloco já foi declarada com mesmo nome num bloco mais externo, a variável ativa no bloco é aquela declarada localmente. A variável *A* deixa de ser global para aquele bloco.

Os exemplos a seguir ilustram variáveis locais, globais e escopo de variáveis.

Exemplo:

```
início
   inteiro: I, J;
   leia (I);
   J ← I ** 2;
   início
      real: X;
      X ← J + 1;
      imprima (X);
   fim;
   leia (J);
   I ← I * J;
fim.
```

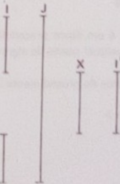
escopo



No exemplo acima, *I* e *J* são variáveis globais e *X* é uma variável *local* ao bloco interno

```
início
   inteiro: I, J;
   leia (I);
   I ← I ** 2;
   início
      real: X, I;
      X ← J + 1;
      imprima (X);
   fim;
   leia (J);
   I ← I * J;
fim.
```

escopo



No exemplo acima, *I* deixa de ser global por ter sido declarada (com o mesmo nome) no bloco interno. Na verdade, é como se fosse uma nova variável *I'*.